5-2015

Persistent Population Biases in Branching Random Walk Algorithms

David J. Geroski
College of William and Mary

Persistent Population Biases in Branching Random Walk Algorithms

A thesis submitted in partial fulfillment of the requirement
for the degree of Bachelor of Arts / Science in Department from
The College of William and Mary

by

David J. Geroski

Accepted for HONORS
(Honors or no-Honors)

Shiwei Zhang, Director

Gina L. Hoatson

Gina Hoatson, Physics

George Rublein, ~~Physics~~ Mathematics

Williamsburg, VA
April 29, 2015

Persistent Population Biases in Branching Random Walk Algorithms

David Geroski, Hao Shi, Shiwei Zhang

Department of Physics, College of William and Mary, Williamsburg, VA 23185, USA

May 12, 2015

Contents

We explore the existence of a residual population bias in Diffusion Monte Carlo algorithms. We develop a model problem, which mimics the single-particle, one dimensional simple harmonic oscillator of unitary ground state energy, to study these biases. We then apply an importance sampling algorithm with a population of a single random walker to calculate the ground state energy and wavefunction of our modeled system in cases of both perfect and imperfect importance sampling. In the former case, we perfectly calculate both the ground state energy and corresponding wavefunction of our system, thus validating our calculations and assumed answer. In the latter case, we show the existence of a residual population bias for populations of small numbers of random walkers. We demonstrate the convergence of our small population calculations and show that we cannot remove this bias through a number of single walker methods. Finally, we demonstrate the overall decay of this residual population bias as the population of random walkers expands, allowing our calculations to converge to the true answer.

Key Words: Diffusion Monte Carlo, Population Dynamics, Random Walk

1 Introduction

Quantum Monte Carlo is a discipline in theoretical condensed matter physics which focuses on using Monte Carlo methods to solve the Shrödinger Equation for different model systems. Most particularly, this method is useful in solving many-body models where analytic solutions can not be derived, found or guessed accurately and where more traditional numerical algorithms take too much computational power to calculate. Quantum Monte Carlo is a catch-all for several classes of algorithms and solutions which use Monte Carlo methods. Our research has focused on working in one of these, Diffusion Monte Carlo.

1.1 Diffusion Monte Carlo

Diffusion Monte Carlo, also known as Green's Function Monte Carlo and Variational Monte Carlo, uses the statistical ensemble method by calling on a population of random walkers to sample our equation space. Given an initial configuration, the population evolves according to the rules of probability, eventually con-

verging to a stable state, which represents the square of the ground state wavefunction in the model system. The ground state wavefunction of the system, as well as the energy of the system, can be directly calculated in this manner. As suggested by the name, this method is derived using the Green's Function, and the Shrödinger Equation can then be solved numerically in integral form.

Beginning in a system of N particles, with state configuration $\vec{r}$, which interact with one another through a potential which is dependent on this state configuration, $V(\vec{r})$, which is generally a sum of one and two-body terms. Then, the n-body Schrödinger equation is written, in appropriate units, as

$$(-\Sigma_i \Delta + V(\vec{r}))\psi_k(\vec{r}) = E_k \psi_k(\vec{r})$$

Where $\psi_k(\vec{r})$ is the wavefunction of the system, and E_k is the energy associated with this wavefunction. From here, add a suitable constant, V_0 so that:

$$V_0 + E_0 \geq 0$$

In this way, the Green's function:

$$-(\Sigma_i \Delta + V(\vec{r}) + V_0)G(\vec{r}, \vec{r}_0) = \delta(\vec{r} - \vec{r}_0)$$

is non-negative, and can be used to transform the Schrödinger equation into the integral equation:

$$\psi_k(\vec{r}) = E_k \int G(\vec{r}, \vec{r}_0)\psi_k(\vec{r}) d\vec{r}_0$$

This integral equation is equivalent to the differential form of the Schrödinger equation, because the Green's function is non-negative, and can be solved as an iterated integral provided that the Green's Function itself

is integrable. Diffusion Monte Carlo algorithms focus on using Monte Carlo integration in order to iteratively solve this equation for the ground state of the system, $\psi_0(\vec{r})$. Our methods thus far have focused on approximating the Green's function by simulating Brownian in the evolution of our ensemble within the relevant configuration space. In our model, this simulated Brownian motion takes the form of a Gaussian centered at the origin, which coincides with the center of the oscillator. Then, our importance function mimics our expectation of the behavior of the particle in either position or momentum space. Thus, for a properly chosen Gaussian, we expect the solution to our sampling algorithm to be exact.

1.2 Population Dynamics

Population Dynamics, as they relate to Diffusion Monte Carlo, define the behavior of the sampling ensemble. This governs the movements of the walkers within the defined potential, any interactions between individual walkers, and dimensions which each walker is defined to have access to. So, defining different population dynamics defines the model which a researcher is working on, from a quantum many-body system to a system in statistical physics. Mathematically, the population dynamics control the definition of the Green's function involved in solving the iterated integral, and, as such, for different Population Dynamics, we expect our answers to converge to different energies and ground state wavefunctions. However, we do not generally expect that different initial population configurations will converge to different final solutions for energy and wavefunction. In this way, the population dynamics govern the calculations made in Diffusion Monte Carlo, rather than the initial configuration of the population.

For a given potential, in DMC, the population dynamics of the system are easy to see. Specifically, the each individual walker will take a step defined by the probability of moving as a free particle. Then, this step will be either resisted or encouraged, and thus shortened or magnified, by the defined quantum potential in the space of the model. In this way, representations of particles are able to evolve in time, and eventually able to stabilize and converge to an effective wavefunction. In single-particle or non-interacting systems, this

process is much less computationally demanding, as the potential does not change with the configuration of the ensemble.

1.3 Importance Sampling

In Diffusion Monte Carlo, importance sampling defines the initial configuration of walker. This idea is one which can greatly reduce the runtime in calculations of model systems, since the population configurations will converge more quickly is the initial state better resembles the steady state. While efficient computation is always a goal in this field, many calculations of complicated model systems would not be possible without the significant runtime reductions afforded by importance sampling. For states which are not known, the researcher tends to direct the sampling to areas of the equation space which correspond to well-known results around the model. The researcher constructs a so-called "importance function," which places different weights on areas of the equation space according to what the researcher deems to be relatively more or less important than other areas of the space. In this way, the researcher is able to apply prior knowledge in directing the sampling within the new system. So, the calculations are then better able to converge to the true solution more quickly, saving computer time. However, importance sampling algorithms are dangerous, since they should only effect the runtime of the calculations, rather than effecting the final answer. Essentially, the researcher should be able to apply his prior knowledge to optimize the calculations and to allow for greater stability in the calculations, but should not be able to change the final answer to a model problem at will.

In Diffusion Monte Carlo, importance sampling takes the form of a "trial wavefunction," or $\psi_T(x)$, which defines the initial configuration of the population. In this case, it is easy to see the benefits of importance sampling. For instance, in making calculations on most model systems, we expect the relevant equation space to be localized in and around our model in space, rather than far removed from it. In this way, we are easily able to place a greater importance on random walkers which are centered near our system.

Also, we can see that if we choose the initial configuration of the ensemble to be exactly the ground-state wavefunction of our system, then the wavefunction of the system will not have to be iterated, but instead will immediately yield the exact solution to the model problem. This allays our fears, as it is easy to see that defining different initial configurations of the system should not effect the final outcome of the calculations, but should only truly effect the time that it takes to make the calculations, except in very special cases.

The only time that we expect our algorithm to converge to a different final solution based on initial population configuration is if the initial population is configured orthogonally to the ground state wavefunction. In this case, we expect to make an excited state calculation. This is still a valid calculation of the system, it is simply not the solution which we expect an actualization of the model to display. This type of situation is generally only seen where the researcher is trying to make an excited state calculation. While other classes of Quantum Monte Carlo are more often used to make excited state calculations of modeled systems, this method would allow a researcher to make this type of calculation using Diffusion Monte Carlo. However, this presupposes that the researcher has an exact knowledge of all lower-energy states of the system, and can then choose an initial population configuration which is mutually orthogonal to all of them. For this reason, Diffusion Monte Carlo is rarely used to make excited state calculations and we do not expect to see this problem while making a ground state configuration calculation.

It should be noted that importance sampling always makes an impact on Monte Carlo calculations. In Diffusion Monte Carlo, we can easily see this since there is always an initial configuration of the population. In model problems which we do not think of as utilizing importance sampling, we would simply see that our algorithm is using an importance function which attaches unit importance to all points in the equation space. However, it is only in algorithms which utilize non-unitary importance sampling that computation time can be greatly reduced. In this way, the algorithm then has a built-in mechanism for spending more time sampling the points that the researcher describes as more critical, as opposed to spending the same amount of time sampling all sets of points. Again, the closer that the researcher is to defining the absolute

ground-state wavefunction of the model system, the faster that the calculation can be performed. This speed can be improved until the situation of perfect importance sampling, wherein the initial configuration only needs to be iterated once to be shown to as have converged to a stable solution.

1.4 Residual Population Biases

Diffusion Monte Carlo calculations ought to converge to the appropriate final solution no matter the size of the defined population of random walkers. In practice, these calculations have always been made using a large population of walkers, because in simple systems this gives researchers more confidence in the calculated solution. However, simply defining a large population of random walkers is not conducive to making large, many-body calculations. Making an M-body calculation in d dimensions, we find that we need a separate population of walkers to calculate the groundstate configuration for each particle in every dimension. So, for a population size of N, we see that the algorithms scales according to:

$$\odot = N * M^d \tag{1}$$

For complex calculations, the timing coefficient can depend on N, M, and d, but this usually carries only a weak dependence. So, we want to minimize M to make the most efficient calculation possible. There is no mathematical reason that a population of a single random walker should not be possible, but when making single walker calculations, a residual population bias always seems to arise. Such a bias would demand a large population of random walkers to handle it.

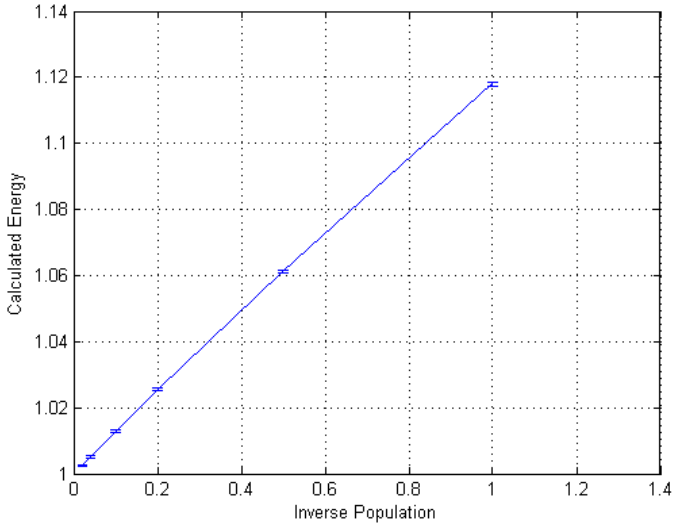

Figure 1: Calculation Showing Population Bias, Small populations of random walkers show a bias, in that they do not converge to the correct solution. Also, the solution the population converges to is outside of statistical noise shown in error bars.

This residual population bias, as it relates to Diffusion Monte Carlo algorithms, is not well understood or characterized within the field. There is, in fact, a long-held belief that this issue simply arises due to the inherent limits of the computational power used in making the calculations, rather than being a source of bias within the numerical method itself. Employing a large population of random walkers, then, becomes a means to an end of dealing with the problem of numerical precision, rather than a means by which a researcher removes this bias. So, while this bias is not necessarily having a major impact of calculations being made, proving that this bias is independent of computational precision and characterizing the bias is worthwhile overall. Following this logic, the goal of our research has been to prove the existence of such a residual population bias, as well as to quantitatively investigate this problem and its causes.

This thesis outlines the use of Diffusion Monte Carlo algorithms to study a simple model problem, which mimics the single-particle, one-dimensional simple harmonic oscillator of unitary ground state energy. In this model, we effectively have infinite computational power and memory, we are able to investigate sufficiently

long timescales and large populations in order to obtain an exact solution to the model. Additionally, giving the simple numbers used, we are able to demonstrate that this bias exists even where we have not lost our precision due to the limits of computation. Working with this model, we show the presence of a residual population bias and begin characterizing it. We then outline and examine several possible solutions to the problem, and demonstrate that the only way to remove this residual population bias is to expand the size of the sampling ensemble.

2 Our Model Problem

Due to the computationally expensive nature of many quantum many-body problems, we have constructed a model problem which can serve to capture the essence of the population biases, but which does not require an unnecessary expenditure of computer time. Additionally, we chose a model problem which mimics the one dimensional simple harmonic oscillator, which has a well known analytic solution. Because of the presence of an analytic solution, we were better able to identify biases in our numeric solutions, rather than comparing our calculations to calculations previously made in the field.

$$\widetilde{G}(y,z) = \sqrt{\frac{\frac{3}{2}+\gamma}{\pi}} e^{-(\frac{3}{2}+\gamma)(y-\frac{z}{\frac{3}{2}+\gamma})}$$

So that the final relevant equations to calculate the energy and wavefunction for the system become:

$$\widetilde{W}(z) = \frac{2}{\sqrt{3+2\gamma}} e^{-(\gamma-1+\frac{1}{\frac{3}{2}+\gamma})z^2}$$

$$\widetilde{\psi}_0(y) = \widetilde{W}(y)\psi_T(y)\psi_0(y)$$

$$\widetilde{\psi}_0(y) = E_k \int \widetilde{G}(y,z)\widetilde{W}(z)\widetilde{\psi}_0(z)dz$$

$$E_k = \frac{\int \widetilde{\psi}_0(y)dy}{\int \widetilde{G}(y,z)\widetilde{W}(z)\widetilde{\psi}_0(z)dydz}$$

When these two integrals are sampled stochastically, these two integrals turn into a discrete sum of their integrands over the numerical trials that are performed.

$$E_k = \frac{\Sigma_i W_i}{\Sigma_i W_i \widetilde{W}_i}$$

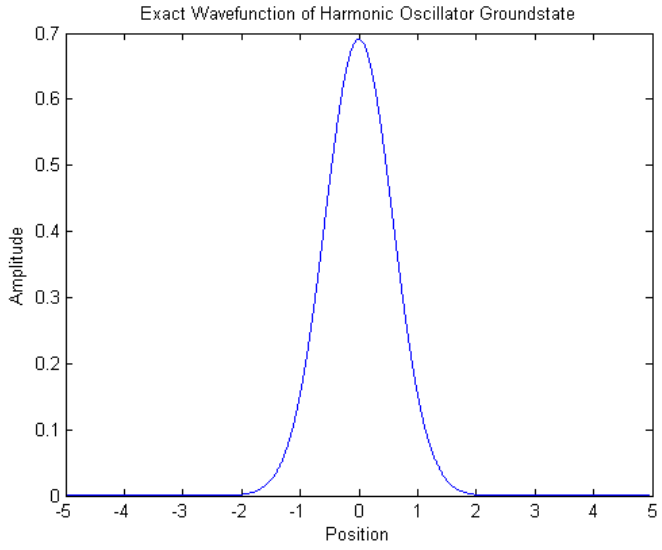

Figure 2: Ground state wavefunction of Simple Harmonic Oscillator

2.1 Importance Sampling in the model

For our model, we had the goal of finding initial population configurations which could easily afford us both perfect and imperfect importance sampling. On top of this, we wanted to define a parameter which would control the importance sampling, and easily define the "goodness" of the importance function which we are using. So, in this case, we chose to sample according to a Gaussian distribution, controlled by the parameter γ. So, we altered the trial wavefunction above, to be written as a weight function multiplied by the initial importance sampling function. In this way, the weight function represents the collective importance of each walker and thus defines the strength of its voice within the overall ensemble, and so walkers are able to gain and pickup weight based on how close they are to the correct solution, and we were better able to understand

the dynamics of the behaviors of individual walkers in our population.

$$\psi_T(z) \to W(z)\psi_T(z)$$

The trial function, $\psi_T(x)$, was then set to sample according to a Gaussian distribution, which is the ground state wavefunction solution to the model problem for an appropriately chosen γ. However, this γ could also be altered to give imperfect importance sampling. This then yielded a weight function, $W(z)$ such that $W(z)\psi_T(z) = \widetilde{W}(z)$

$$\psi_T(z) = e^{-\gamma z^2}$$
$$W(z) = \frac{2}{\sqrt{3+2\gamma}} e^{(-1+\frac{1}{\frac{3}{2}+\gamma})z^2}$$

Mathematically, this alteration is done by left multiplying the importance, or trial, wavefunction to both sides of the integral equation, and then by placing the identity wavefunction in the integral equation. Note that since the energy of the configuration is a constant, it will commute with the importance sampling operator.

$$\psi_k(\vec{r}) = E_k \int G(\vec{r}, \vec{r}_0)\psi_k(\vec{r}_0)d\vec{r}_0$$

$$\psi_T(\vec{r})\psi_k(\vec{r}) = E_k \int \psi_T(\vec{r})G(\vec{r}, \vec{r}_0)\psi_k(\vec{r}_0)d\vec{r}_0$$

$$\psi_T(\vec{r})\psi_k(\vec{r}) = E_k \int \psi_T(\vec{r})G(\vec{r}, \vec{r}_0)\psi_T^{-1}(\vec{r}_0)\psi_T(\vec{r}_0)\psi_k(\vec{r}_0)d\vec{r}_0$$

Then, letting $\widetilde{G}(\vec{r}, \vec{r}_0) = \psi_T(\vec{r})G(\vec{r}, \vec{r}_0)\psi_T^{-1}(\vec{r}_0)$ and $\widetilde{\psi}(\vec{r}) = \psi_T(\vec{r})\psi_k(\vec{r})$, we see:

$$\widetilde{\psi}(\vec{r}) = E_k \int \widetilde{G}(\vec{r}, \vec{r}_0)\widetilde{\psi}(\vec{r}_0)$$

This form is mathematically exact, and the greater control that it offers the researcher can be seen in the fact that, for importance functions which are close to the wavefunction under examination, the integral over the configuration space approaches unity, so that:

$$\psi_T \approx \psi_k \Rightarrow E_k \int \widetilde{G}(\vec{r}, \vec{r}_0) d\vec{r}_0 = 1$$

Again, this method is mathematically exact, so that any invertible trial function operator applied in this transformation will approach the same answer. However, the "goodness" of the trial function can greatly alter the convergence time involved in an algorithm. On top of this, we expected to see, based on previous research in this topic, drastically large deviations from truth in both wavefunction and energy calculations as our importance sampling got worse. It should be noted that the solution obtained in this numerical algorithm can then be transformed back to space with unitary importance, if so desired.

3 Results

Our work this year has focused on proving that a residual population bias must be considered as a factor in solving model systems using Diffusion Monte Carlo. Following previous work in the field, we wanted to validate our calculations by proving that there is no residual population bias for systems calculated using perfect importance sampling. Then, we deviated from this perfect importance sampling and demonstrated that any small deviations from perfection in importance sampling led to a bias in our calculations. We have also worked to begin qualitatively and quantitatively examining these biases, and how residual population biases behave under different importance sampling functions. Specifically, we wanted to make a comparison between different importance sampling functions and show how the "goodness" of the importance function affects the residual population biases.

3.1 Single Walker Population with Perfect Importance Sampling

Given previous work in the field, our first step was to demonstrate that a single walker population demonstrates no residual population bias in models with perfect importance sampling. Upon running this test, with $\gamma = 0.5$, we were able to make an exact calculation of the ground state energy and wavefunction. From the first time step, the walker returned unitary energy for the system and this answer did not waver. Additionally, we were able to reproduce the ground state wavefunction which we were expecting.

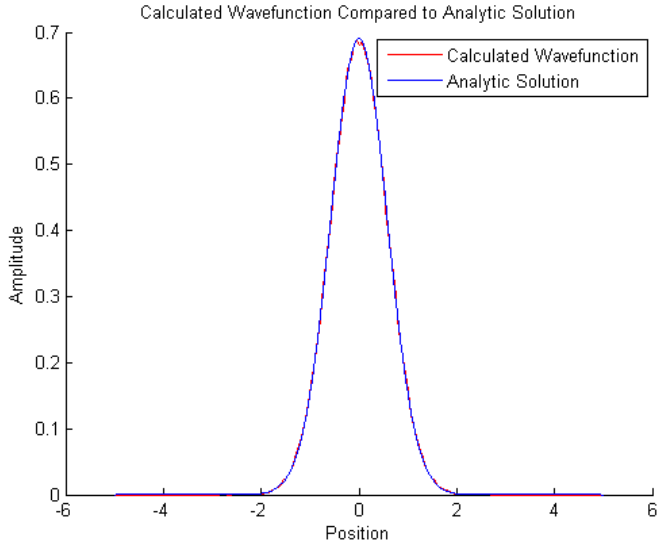

Figure 3: Calculated Ground State Wavefunction, calculated with a single walker using perfect importance sampling

Figure 3 shows the results of this calculation, with the analytic solution to the ground state of the Simple Harmonic Oscillator superimposed on top of our calculated wavefunction. In this case, we are able to see the two different wavefunctions exactly overlapping. Additionally, the walker showed no overall loss in weight or deviation in energy calculation over any length walk. So, we accomplished the dual goal of validating our calculation, by showing it produces the results we said it would, and of demonstrating that residual population biases go to 0 in situations of perfect importance sampling.

3.2 Single Walker Population with Imperfect Importance Sampling

Next, we moved on to make several similar calculations using imperfect importance sampling. In each case, we noticed that single walker calculations not only failed to converge to the energy calculated using perfect importance sampling, but also failed to converge to a single answer in separate threads.

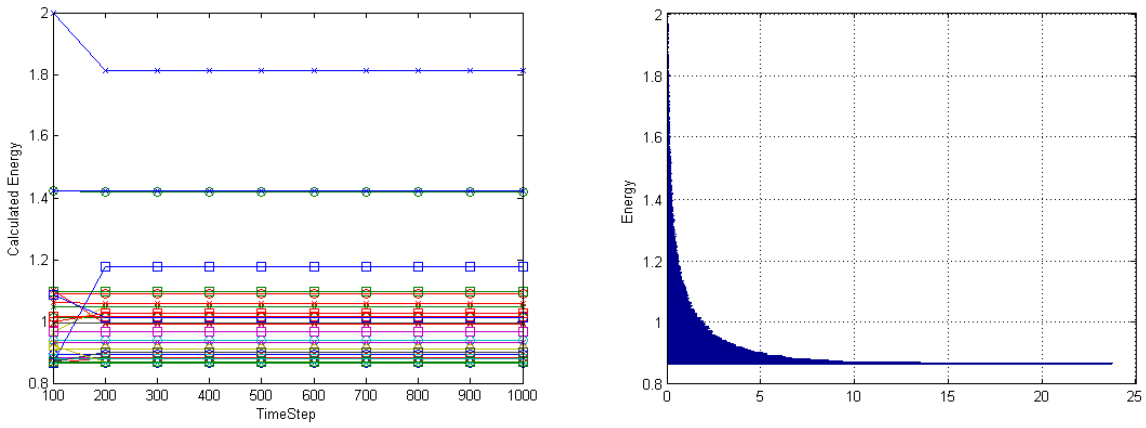

Figure 4: Single Walker Calculations for $\gamma = 0$, the first figure (left) shows the progression and convergence of 20 single walker threads over time, and the second figure (right) shows the probability density function of single walker calculations.

Figure 4 displays 20 individual walker energy calculations, tracked over time and according to weight. All trials display the same essential idea. Most of the walkers converge to an energy value just below the true value. Almost all other walkers converge to values above the true energy value, and the spread above true energy value is much looser than the spread below. Also, of the million walkers employed to show this distribution, only a very small fraction of them converge to the true value.

We see this pattern with every importance sampling function that we use. However, as the importance

sampling function approaches perfect importance sampling, the spread of the single walker threads gets smaller and approaches a δ-function centered around truth, which is only achieved in situations of perfect importance sampling. However, for any imperfection in the trial function, there is always a spread in the energy calculations made by single walkers.

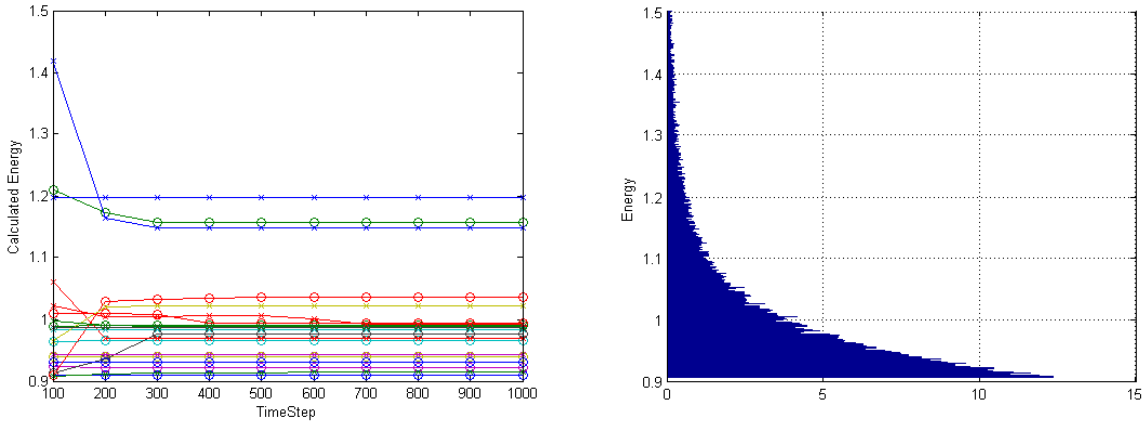

Figure 5: Single Walker Calculations for $\gamma = 0.15$, the first figure (left) shows the convergence of 20 single walker threads over time, and the second figure (right) shows the probability density function of single walker calculations for this importance function.

So, we can clearly see that single walker calculations do not necessarily converge to true energy values, and, in fact, they usually converge to a value just below truth. On top of this, the shape and spread of the values calculated by single walkers changes drastically with different importance sampling schemes. Hence, all single walker calculations seem to be biased in some way, and converge to the wrong solution in almost all cases.

3.3 Exponential Decrease in Weights

Our next step was to demonstrate that these single walker calculations have actually converged. While the calculated energy values do not seem to be changing any more by the end of the walk, it is possible that each one is still moving towards truth infinitesimally, and will eventually converge to the exact solution. So,

we examined the overall weight of the population, and found that the collective weight of each population fell off exponentially unless perfect importance sampling was employed.

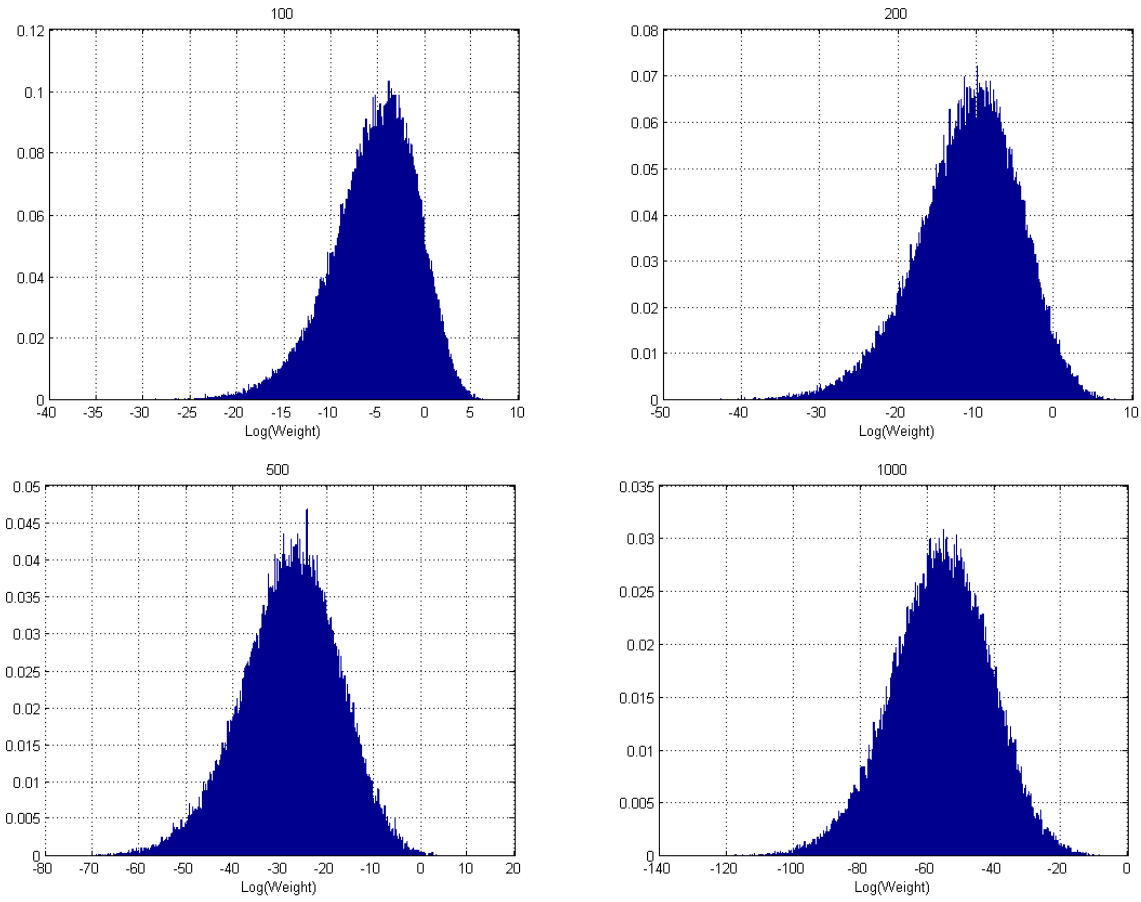

Figure 6: Probability Density Functions of the Logarithm of Population Weight for $\gamma = 0$, Tracked Over Time for timestep $100, 200, 500$, and 1000

We saw the weights of all walkers decreasing as the walk increased. Although some walkers initially pick up weight, over time all walkers eventually lose weight. The form of the logarithm of the Probability Density Function is approximately that of a Gaussian which is slowly shifting to the left and spreading out. Also, over time, the Gaussian distorts, with lighter walkers being more prominent in the distribution.

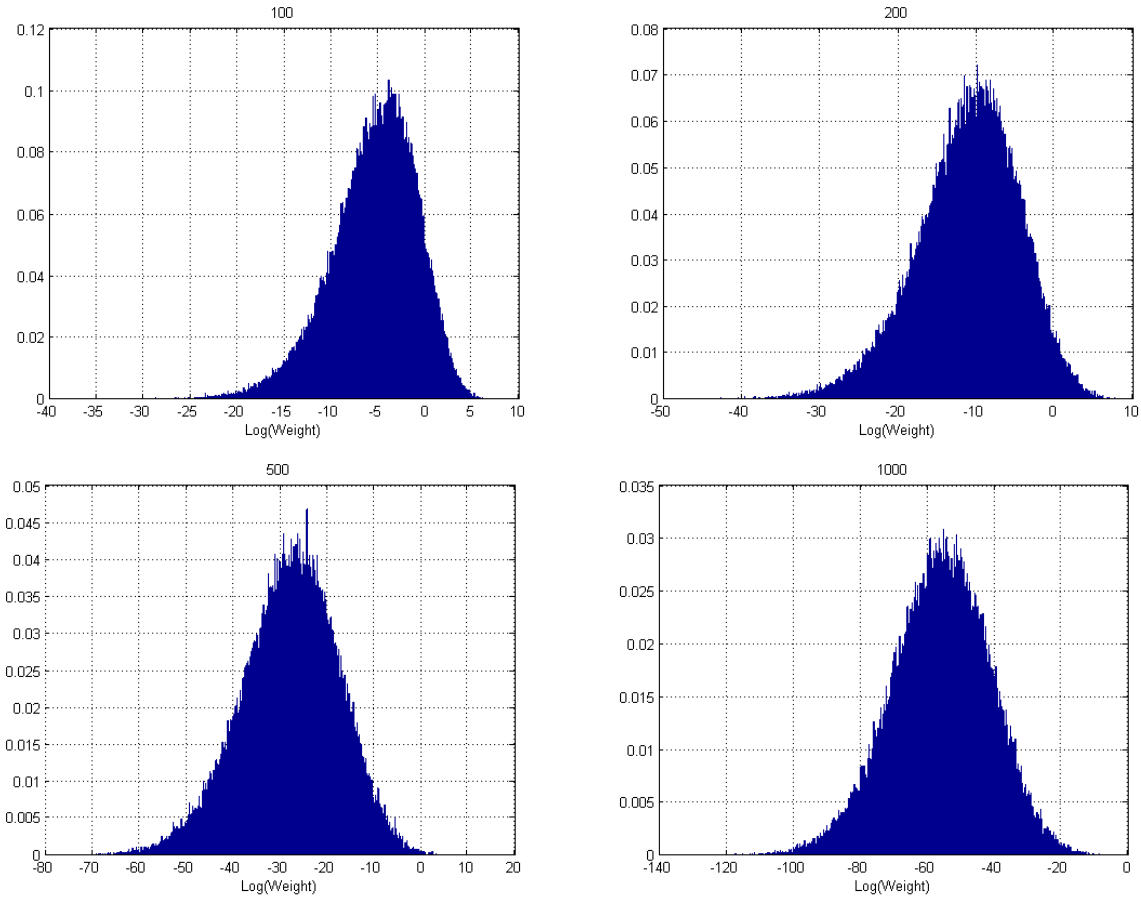

Figure 7: Probability Density Functions of the Logarithm of Population Weight for $\gamma = 0.15$, Tracked Over Time for timestep $100, 200, 500$, and 1000

The shifting Gaussian of the weight distributions can be understood, as the overall weight of a walker is simply the multiplication of all instantaneous weights picked up over the course of its walk. Each of these instantaneous weights, as mentioned previously, is essentially the exponential of a Gaussian. Therefore, the logarithm is essentially the summation of several Gaussian distributions, yielding the distorted Gaussian which is our probability density function.

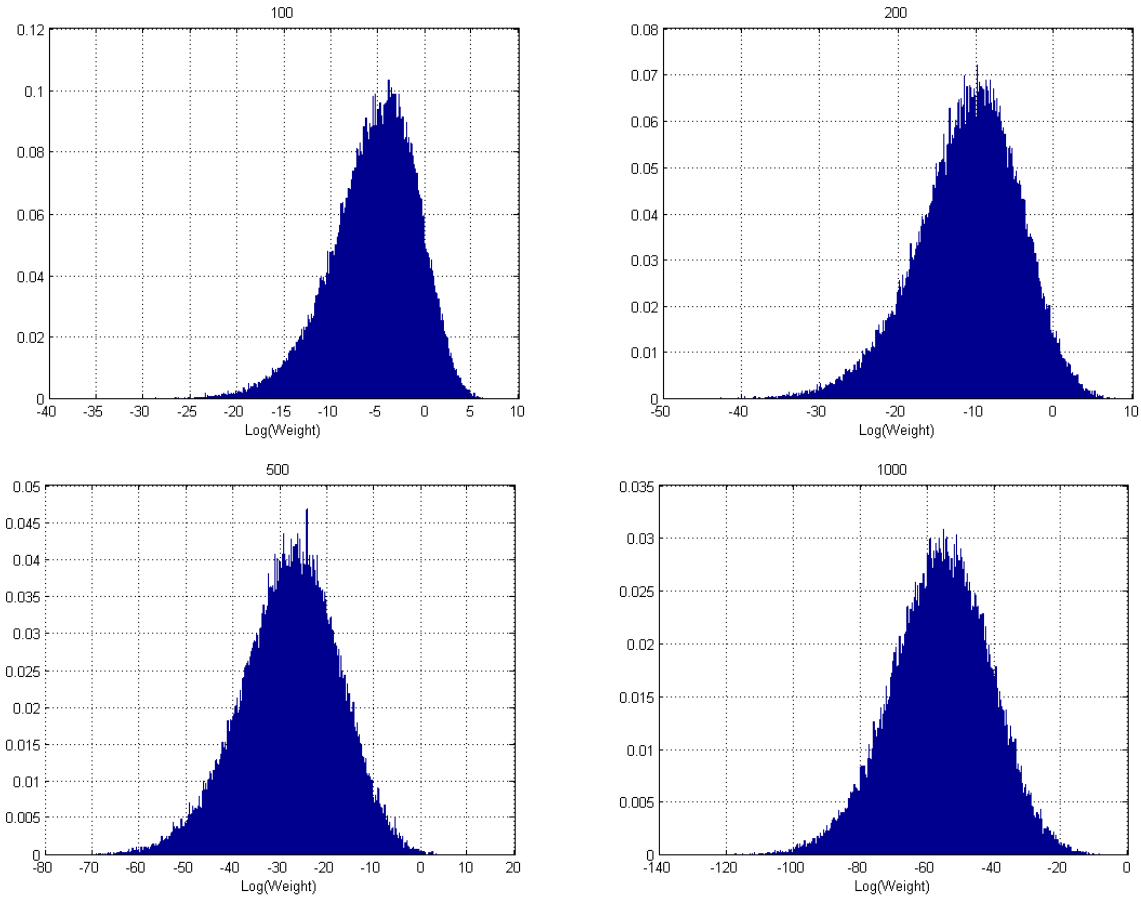

Figure 8: Probability Density Functions of the Logarithm of Population Weight for $\gamma = 0.35$

After comparing the change in each collective population's weight, we compared the collective weight of each population in comparison to one another. We found that better importance sampling schemes led to walkers losing weight less slowly overall, with the situation of perfect importance sampling losing no weight over time.

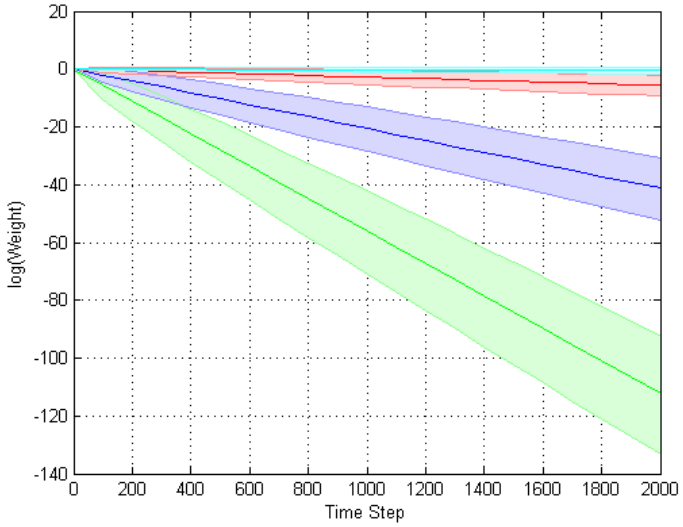

Figure 9: Collective Population Weight for Imperfect Importance sampling, shows, in order from highest to lowest: $\gamma = 0.5, \gamma = 0.45 \gamma = 0.35, \gamma = 0.15, \gamma = 0$ Overall decay of collective weight for any imperfect importance sampling function.

In Figure 9, we clearly see the same exponential decrease in weight with respect to time as we did in the individual functions, with the individual weights of nearly all walkers going to 0 as time continues. On better importance sampling schemes, we can see that some walkers are initially picking up weight as the walk continues. However, as the collective weight of the population decays, so too will the weights of these walkers. Given our way of calculating collective energy,

$$\tilde{E} = \frac{\Sigma_i w_i}{\Sigma_i w_i \widetilde{W}_i}$$

and given the values that we are expecting, $\widetilde{W}_i \tilde{=} 1$ and $w_i \tilde{=} 1$ at the beginning of the walk and then exponentially falling off, we can definitively say that the individual walker calculations have converged to within a value of 10^-40 away from their final value in the case of unitary importance sampling. Specifically examining the plot for $\gamma = 0.15$, we can see that by time step 2000, we expect to be adding values which are (number) orders of magnitude smaller than the initial step, so that it is not making any noticeable difference on the

collectively calculated values. Additionally, we expect that this value will continue to go down as the length of the walk is increased.

In the past, it has been thought that the residual population bias only arose as a response to a loss of numerical accuracy as weights got smaller. Initially, we resolved to compensate for this problem solely by examining a small problem. However, we also strove to compensate for this by tracking the logarithm of our weight, thus allowing us to not lose numerical precision for any walk that we might calculate. Despite these corrections, which we used to eliminate the problem of losing numerical precision, the residual population bias clearly remained in our calculations of single walker populations. The cause of this bias is clear; for calculations made with imperfect importance sampling, the weight of nearly all walkers goes to 0 over time. While this is useful, because it allows our calculations to actually converge, it means that as the walk continues, the newly calculated values will have a smaller effect on energy which is calculated overall. So, even if the individual walker does eventually converge to the appropriate ground-state wavefunction, the lack of emphasis placed on the later stages of the walk lead to this population bias rising.

3.4 Unweighted Energy Histograms

Next, we examined the spread of single walker calculations, to see if there was some easily understood spread to the different energy values. Specifically, we wanted to see if we could determine upper and lower bounds to the calculated values and, in doing so, demonstrate that single walker calculations are good for providing some kind of an estimate of the answer to a model system.

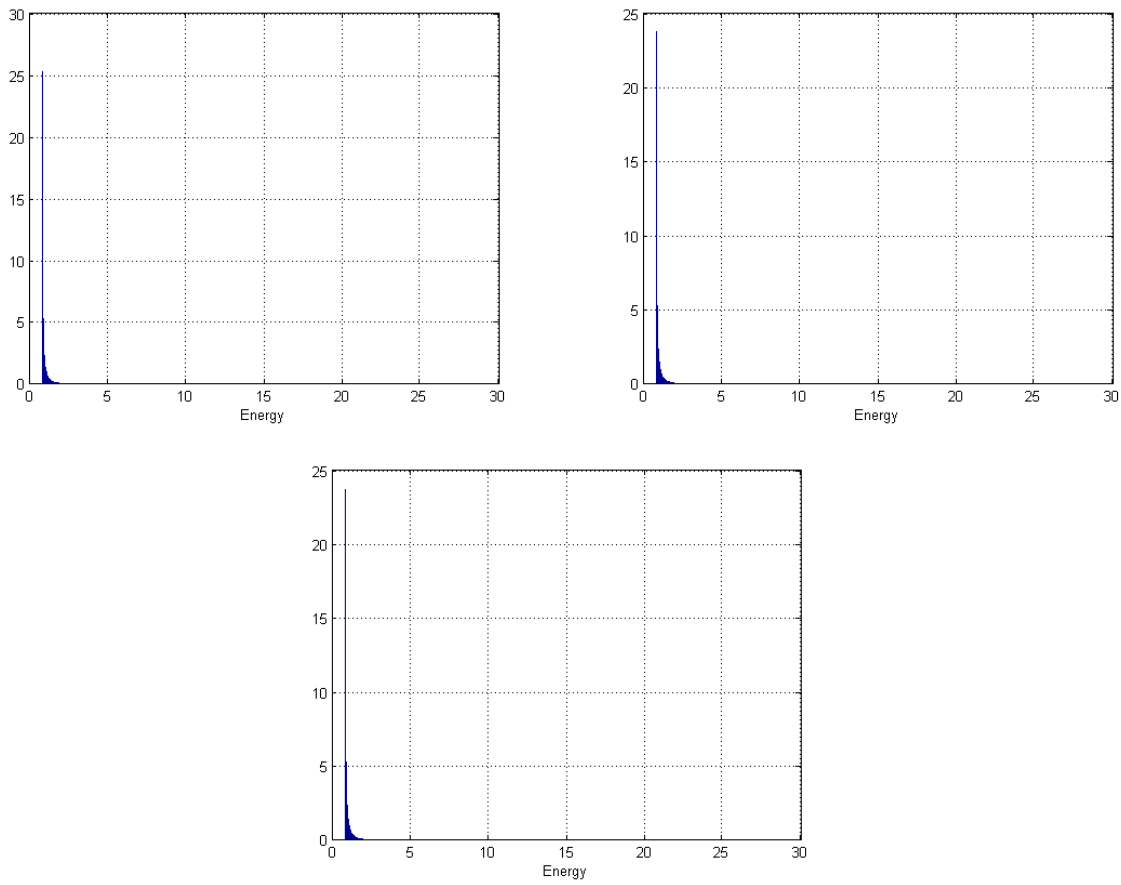

Figure 10: Energy Probability Distribution Functions for $\gamma = 0$ for timesteps $200, 500$ and 1000

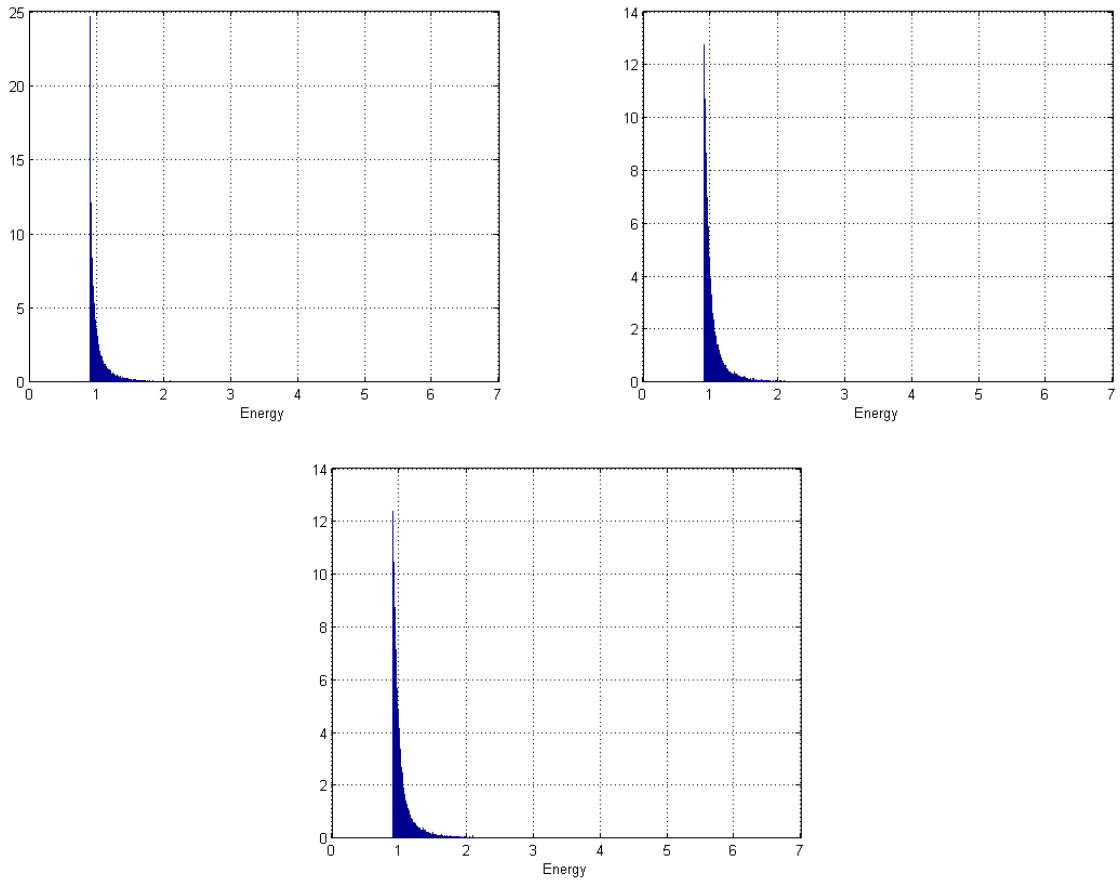

Figure 11: Energy Probability Distribution Functions for $\gamma = 0.15$ for timesteps $200, 500$, and 1000

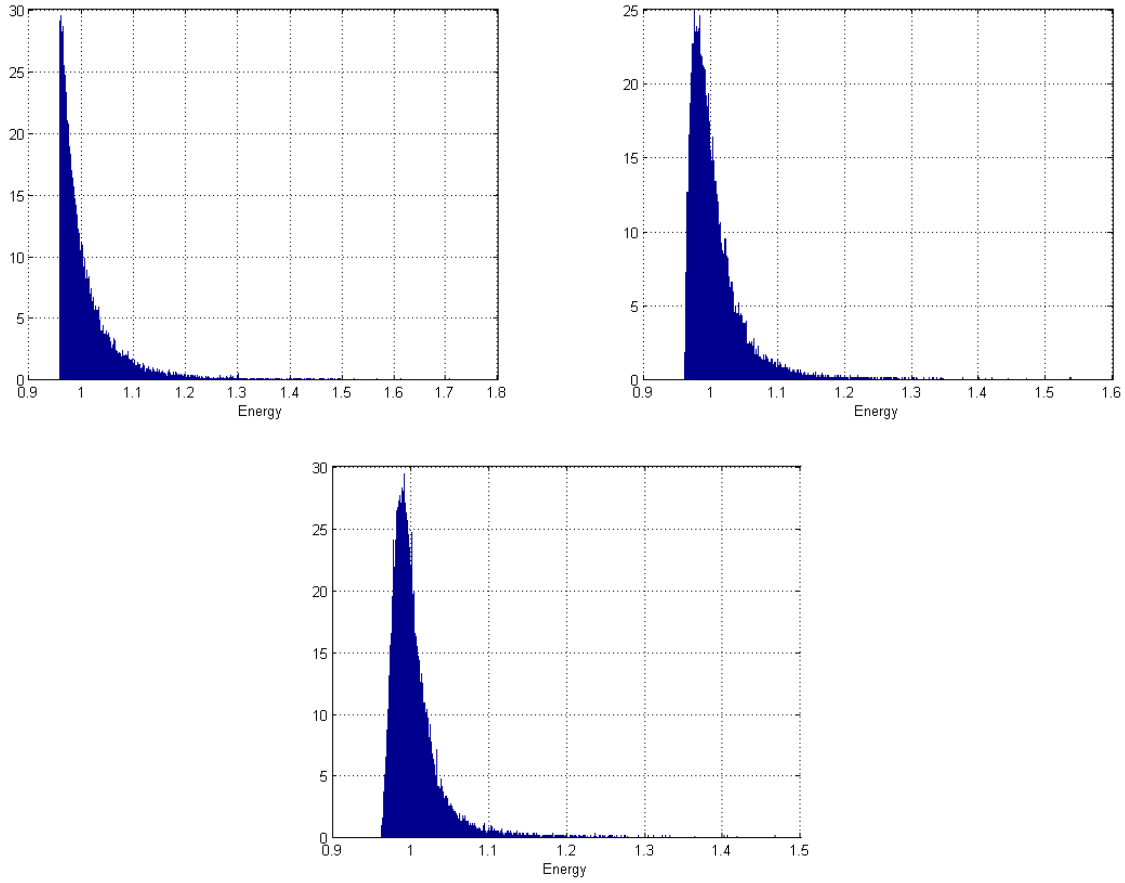

Figure 12: Energy Probability Distribution Functions for $\gamma = 0.35$ for timesteps $200, 500$ and 2000

In all cases, we see a similar spread, which matches the spread that we predicted when we were just examining the single-walker threads. Almost all single walker populations converge to energy calculations which sit just below the true value of the energy, and these values fall off as they increase, ending in a long tail which goes to 0 well above the true value. So, while we are able to say that most single-walker calculations will yield an energy value just below that of the system, there is no guarantee that this will be true, and an individual walker may return an energy value which is well above the calculated value, so an estimate already cannot be made without employing several different walkers in the calculation.

Also, taking average values of these histograms always leads to a calculated value just above truth. Again,

better importance functions lead to values which are closer to truth, which comes part and parcel with the fact that the spread tightens up as we approach perfect importance sampling. So, an unweighted histogram will not yield the true answer to a model system unless the researcher happens to employ perfect importance sampling.

3.5 Weighted Energy Histograms

After examining unweighted histograms and determining their inadequacy in removing the residual population bias, we began taking into account the collective weight of the population. Here, we expected to begin to see the exact solution coming out of our calculations. We found that, as the walk is continued, energy values far away from truth go to 0 more rapidly than the spread around truth.

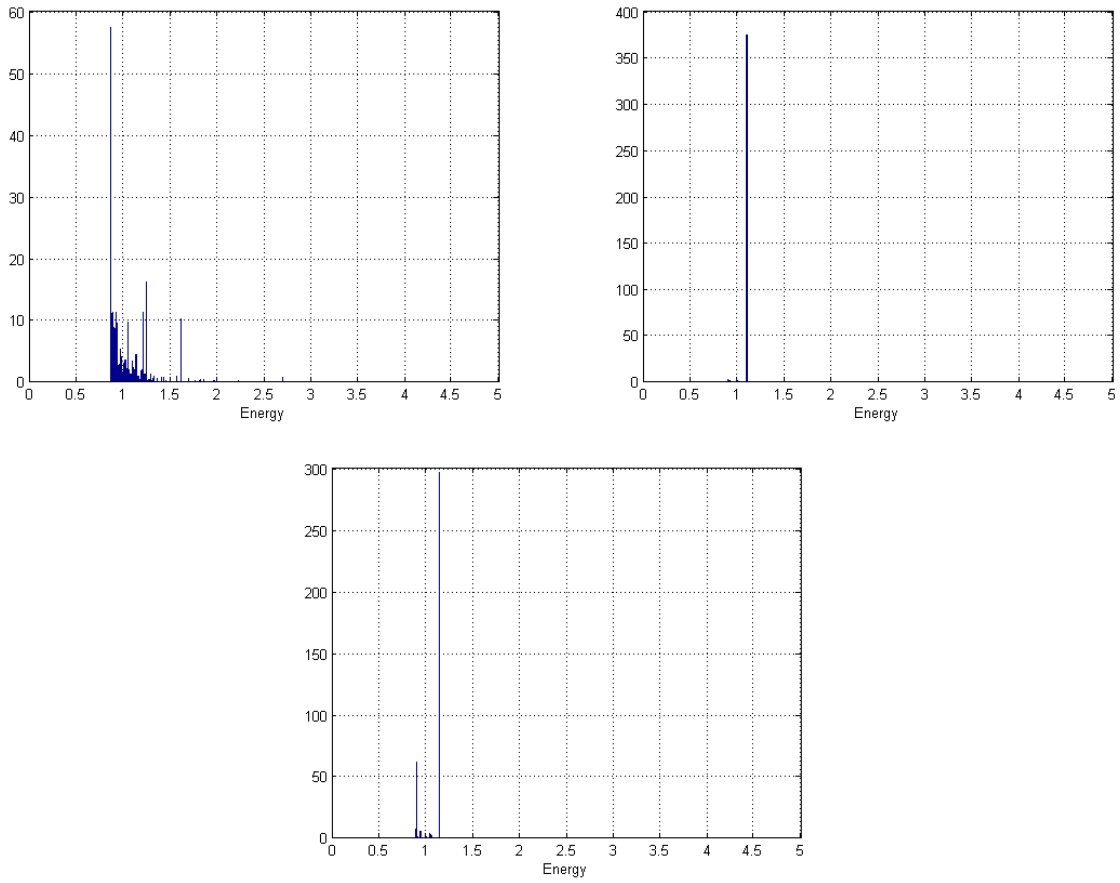

Figure 13: Weighted Energy Histogram for $\gamma = 0$, shows values of approximately 0 for any value away from the true energy.

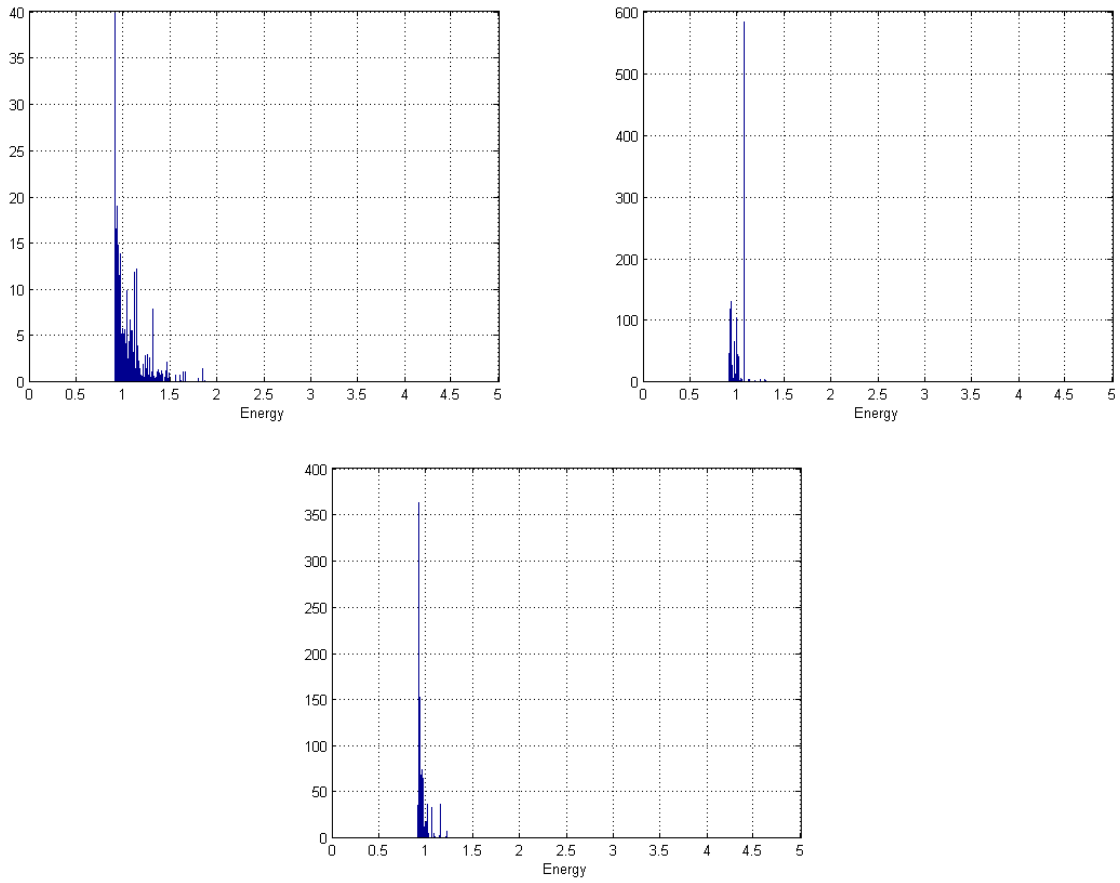

Figure 14: Weighted Energy Histogram for $\gamma = 0.15$, shows no weight for walkers calculating energy values far from truth

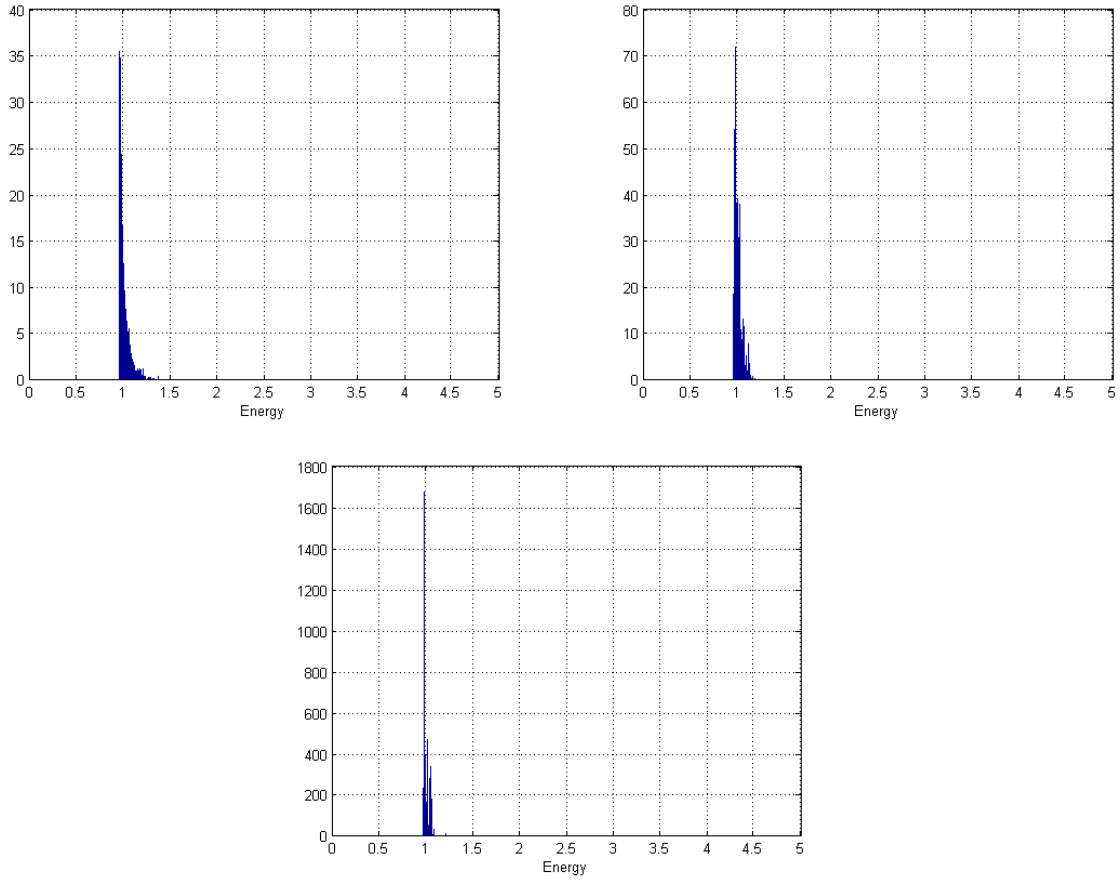

Figure 15: Weighted Energy Histogram for $\gamma = 0.35$, calculating average of weights yields true energy, as expected

Figures (13, 14, and 15) demonstrate all different γ values losing weight at all energy values except those very close to 1. All averages taken of weighted energy values also returned an energy value which was exactly truth. This is because, although the collective weight of the population is low, each walker carries a weight which is proportional to how close it is to truth. In this case, the only energy values which pick up any weight at all are very close to truth and they pick up weight such that the overall calculation approaches truth.

3.6 Inverse Population Plots

Finally, we wanted to definitively demonstrate that the way to remove this residual population bias is by expanding the population and making calculations based upon a much larger population. So, we made a calculation of our model system's energy based on a population of a million walkers. For each γ value, we then examined progressively larger groups of walkers and the spread of their calculated values and found the same as we did before. Small groups of the walkers or small populations of walkers consistently yielded biased values. On top of this, the true energy value was well outside of our standard error bars. However, as we began examining larger populations of random walkers, our values would approach truth. However, all finite population sizes remained biased, and we would only get our calculation close to the true value. So, the only solution was to extrapolate to an infinite population size.

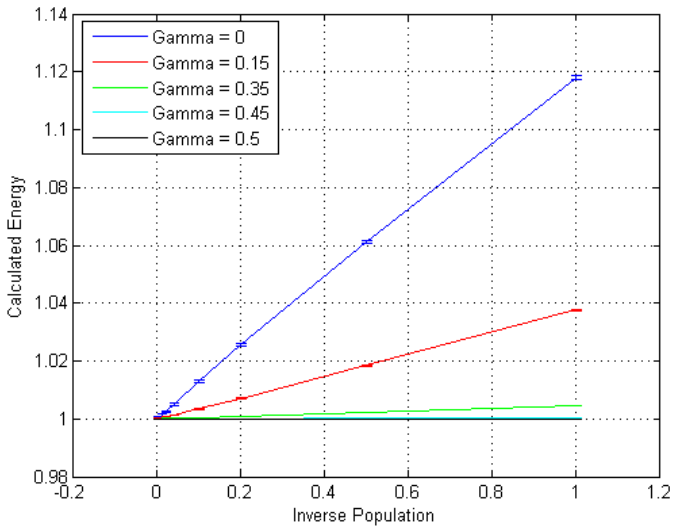

Figure 16: Expanded Population Calculations for several importance functions

As we can clearly see in Figure 16, we are able to linearly extrapolate to the correct result in cases of infinite population size for any importance sampling function. With this, we had conclusively demonstrated that we are indeed looking at a residual population bias. This bias exists for any population size, even

though it can be made arbitrarily small with larger populations, and it can only be reduced by expanding the population size or by guessing a better solution for the importance function than has been done previously. In fact, the only way to truly remove this bias was to extrapolate to infinite population size, or to guess the correct answer. It is also worth noting that if we were to make the same calculation without taking into account the weights of each walker, then every calculation would converge to the single-walker limit, shown for an inverse population value of 1.

3.7 Weight Compensating Algorithm

We ran one final test. Since we discovered that the algorithm is converging to the wrong solution due to the loss of walker weight, we tried compensating for this loss of weight over the length of the walk. In this way, we would still be able to make a single walker calculation, but there would be more emphasis placed on later steps in the walk. As previously shown, we could not simply discount the weight, as the walk would then not converge to the right solution for any population size. So, we could reset the weight only periodically. Another concern is that we could not change the calculation such that we were effectively making a multi-walker calculation. For this, we decided to simply reset the weight of the walker to unity periodically. Additionally, we could reset the position of the walker, to make sure that it stayed on track. However, we could not re-thermalize the walker, since this would be equivalent to using multiple walkers, and we certainly could not reset the walker's energy calculation, since this would again be equivalent to using multiple walkers. So, we made two calculations, one where we only compensated for weight, and one where we periodically placed the walker back in the center of our oscillator.

To make the overall weight compensation, we decided to periodically reset the walker weight to unity, as it was at the beginning. These initial calculations, predictably, showed a much longer time for convergence than our earlier single walker calculations. Also, this single walker calculation took a much longer time to converge than it would have taken to use a large enough population to make our calculation, removing all of

the reasoning behind using a single walker to make a calculation. Then, we still found that all of the walkers converged to different final energy values, and that they didn't converge asymptotically to the true solution.

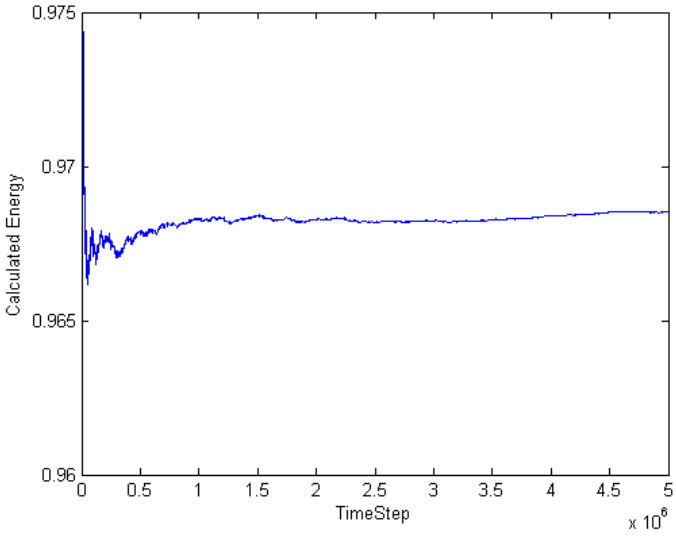

Figure 17: Single Walker Calculation employing Weight Compensating Scheme, shows closer convergence to truth, but still displays bias.

Our initial calculation shows the invalidity of this scheme in Figure 14. With our single walker calculation converging to $E = 0.968$, rather than to $E = 1$ as we desire, we can see that this single walker calculation converges to the wrong solution. Additionally, in the time span that it has taken to make this calculation converge, we could have made an expanded population calculation of $25,000$ walkers. We then followed up this calculation with several more similar calculations, to see if all of the threads would converge to the same value. Figure 15 shows that the threads fail to converge to the true value, and also that they all converge to different values.

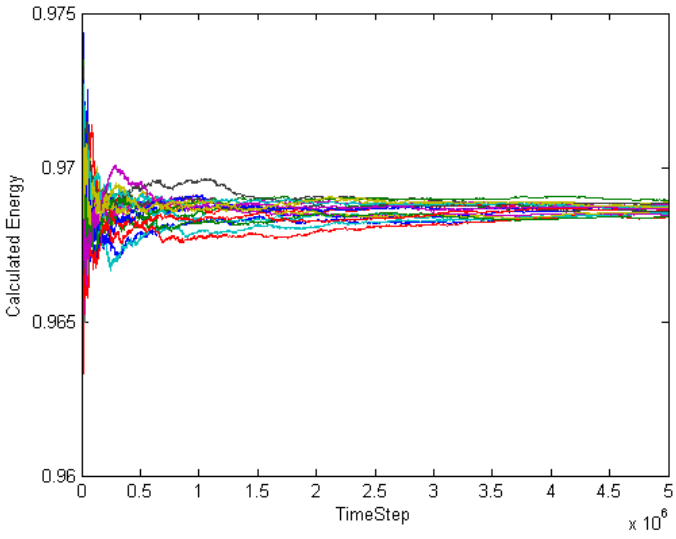

Figure 18: Ten Single Walker Calculations employing Weight Compensating Scheme, converge to values which are closer to one another and closer to truth, but do not converge to one another exactly or to truth exactly

After making the initial weight compensation calculation, we then made another calculation in which we not only periodically reset the weight, but at the same time reset our walker's position back to 0, the center of our simple harmonic oscillator. This calculation showed all of the same problems as the previous one, in that it took longer to converge than other calculations. Figure 16 shows our initial attempt, where the calculation converges to approximately $E = 0.9675$

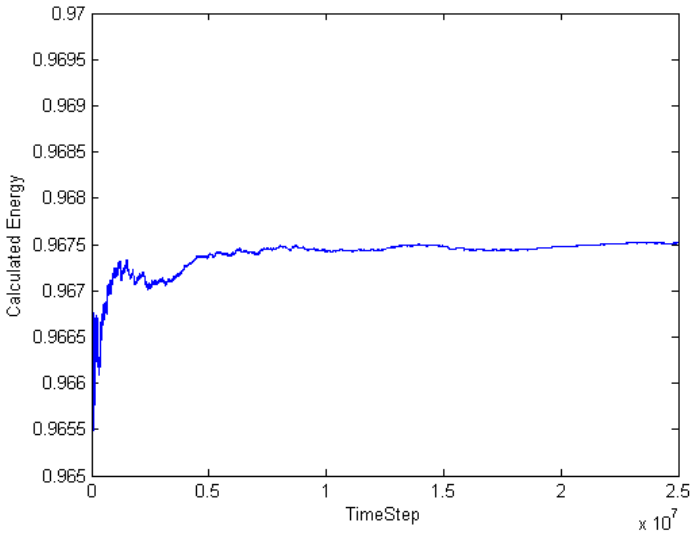

Figure 19: Weight and Position Compensating Thread

Again, taking the next step of re-thermalizing the walker at each stage of the walk would be equivalent to using multiple walkers. So, we concluded that this weight compensating scheme was inadequate for removing this residual population bias. We ran several more weight and position-compensating calculations to see the final spread of calculations made using this scheme. Predictably, they ended up with a smaller final spread than any other single walker calculations, but this is because the walks were, overall, more often reset to be nearer to one another. The values all converged between $E = 0.967$ and $E = 0.968$

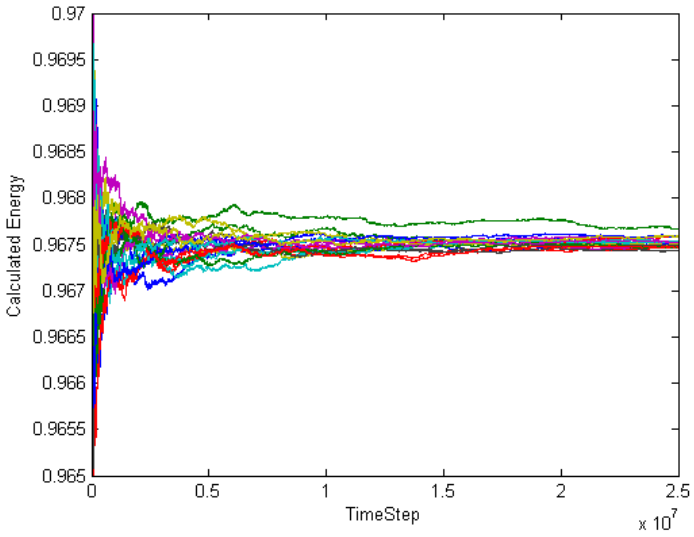

Figure 20: Twenty Weight and Position Compensating Threads

4 Conclusion

We have outlined the problem of a residual population bias in Diffusion Monte Carlo algorithms. We studied this problem and a number of potential solutions to it by setting up a model problem. Our model problem mimicked the single-particle, one-dimensional simple harmonic oscillator, and was chosen because it is analytically solvable and has a well defined energy and ground state wavefunction. We then showed that this residual population bias arises even in this relatively simple problem, and demonstrated that it exists unless the system has perfect importance sampling.

First, we demonstrated that Diffusion Monte Carlo calculations display no bias for any population size in cases of perfect importance sampling. However, given that this would not allow researchers to study any new problems to which the solution is not known exactly, we decided that this was relatively non-scientific. Then, we demonstrated that single-walker calculations are always biased where imperfect importance sampling schemes are employed. Also, we were able to show the overall decay of this population bias as the

importance sampling is improved.

Additionally, we were able to demonstrate one of the root causes of this residual population bias. This cause is the exponential decay of the population's collective weight. Thus, as individual walker weights tend to decay exponentially to 0 over the walk, the later steps in the walk stop contributing to the collectively calculated energy of the walker. So, even if the walker does eventually sample the entire relevant equation space, the later steps are de-emphasized as the walk goes on.

Finally, we were able to demonstrate that this residual population bias is not, as was previously thought, a result of the loss of computational precision. We have been able to demonstrate this by removing the issue of numerical precision by tracking the logarithm of walker weights and working in a computationally inexpensive model. Further, we demonstrated that, even when compensating for the overall loss of walker weight, there is no way to remove the bias without effectively expanding to a larger population of walkers. Given these factors, we were able to show a convergence of the calculations without any loss of computational precision. Given that these calculations still show a bias, we were able to determine that our residual population bias is independent of a loss of computational precision.

So, we have explored the existence of a residual population in Diffusion Monte Carlo calculations. We showed that this bias exists in even the simplest of systems, such as the simple harmonic oscillator in our model problem, and demonstrated that this population bias exists for any importance sampling function. Additionally, we showed that this bias continues to exist no matter how we account for the sum of the walk over time, and that the bias arises due to an overall loss in collective weight, which is the same reason that the walk converges. We then showed that this bias exists in spite of having near infinite computational precision, power, and memory, as we did in this model system. Finally, we showed that there are only two ways to truly remove this bias: to make the calculation with an infinite population of random walkers, or to make the calculation with perfect importance sampling.

5 References

1. Hansen, J.P., "Monte-Carlo Methods in Classical Statistical Mechanics." *Lecture Notes in Physics 240, Monte-Carlo Methods and Applications in Neutronics, Photonics and Statistical Physics* 153-159 (1985)

2. Kalos, M.H. "Exact Monte Carlo for Few Fermion Systems." *Journal of Statistical Physics* 63.5-6 (1991): 1296-281. Web

3. Kalos, M. H. and Whitlock, Paula A. *Monte Carlo Methods* John Wiley & Sons, 1986

4. Kalos, M.H. "Monte Carlo Calculations." *Supercomputing Applications and Techniques Workshop* January 7-11, 1991.

6 Acknowledgements

We would like to acknowledge the Roy R. Charles Center for Honors Research at the College of William and Mary, we well as Peter Bracken, Monica Deaver, Horace Crater, Janet Jennings, and Julia Phillips, who have generously contributed to this research. Additionally, we would like to acknowledge the William and Mary Physics Department for providing additional resources for our research throughout the year. Finally, I would like to thank my advisor, Shiwei Zhang, and Hao Shi for their constant help throughout this project and their knowledge in helping me see it through to the end.